

Exercices notés, pondération 10%

Consignes

1. Les réponses aux exercices doivent être clairement identifiés par leur numéro.
2. Le fichier doit être remis au format sql, et doit être nommé : VotreNom_Prenom.sql
3. Les fichiers remis dans un autre format ne seront pas corrigés.
4. Au début de votre fichier, nous devons retrouver votre nom et votre prénom.
5. Aucun travail remis en retard ne sera accepté.
6. À remettre dans la boîte de remise ColNet.

Exercice 1(les vues)

1. Utiliser la base de données de votre TP1 pour créer la vue Vbuts qui va contenir tous les joueurs ayant marqué des buts. Cette vue doit être ordonnée par ordre décroissant des buts. Les colonnes de la vue seront : le nom et le prénom du joueur, le nom de son équipe, le nom de la division de l'équipe et la somme des buts marqués.
2. Utiliser la vue Vbuts pour afficher le nom et le prénom des joueurs ayant le même nombre de buts que le joueur MARKOV ANDRÉ.
3. Utiliser la vue pour afficher le nom, prénom, le nom de l'équipe du joueur ayant le plus petit nombre de buts.
4. Utiliser la vue pour afficher le nom, le prénom, le nom de l'équipe du meilleur buteur.

Exercice 2(les vues)

Quatre personnes : Simba, Remi, Coluche et Saturne ont évalué chacun 10 restaurants. Ce que nous souhaitons avoir est :

1. La moyenne des étoiles pour chaque restaurant.
2. La moyenne des évaluations pour la qualité du service pour chaque restaurant.
3. La moyenne des évaluations pour la qualité de la bouffe pour chaque restaurant.
4. Le meilleur ou les meilleurs restaurants selon la moyenne des évaluations.

D'abord, il faudra exécuter les requêtes suivantes pour prendre connaissance des évaluations de chacune des personnes :

```
select * from simbaresto;  
select * from remiresto;  
select * from saturneresto;  
select * from colucheresto;
```

Questions : Écrire les requêtes suivantes :

1. Utilisez l'opérateur UNION ALL pour créer la vue Vrestaurants qui va contenir l'ensemble des évaluations (des quatre personnes).
2. Utilisez la vue pour écrire la requête qui affiche le nom du restaurant, la moyenne des étoiles, la moyenne pour la qualité du service, la moyenne pour la qualité de la bouffe.
3. Afficher ensuite le nom du meilleur restaurant (celui qui a le plus d'étoiles selon la moyenne).

Exercice 3 (information hiérarchisée)

Utiliser la table (synonyme) SYEMP. Faire d'abord un SELECT * sur cette table

1. Écrire une requête qui affiche les subordonnés de JONES.
2. Écrire une requête qui affiche les supérieurs hiérarchiques de l'employé JONES.
3. Quel est le salaire moyen des employés supérieurs de JONES ?

Exercice 4 (information hiérarchisée)

En ce temps de confinement, nous vous demandons de rester proche de votre famille. De discuter de vos grands-parents, arrière-grands-parents...

On vous demande de remonter jusqu'à votre 5^{ème} arrière-grand-père pour construire votre arbre généalogique de vos grands-parents.

Le code SQL correspondant est le suivant :

```
CREATE TABLE genealogie (
  no_enfant      NUMBER(3) NOT NULL,
  nom_enfant     VARCHAR2(30) NOT NULL,
  prenom_enfant  VARCHAR2(30) NOT NULL,
  profession_enfant VARCHAR2(50) NOT NULL,
  no_papa        NUMBER(3)
);

ALTER TABLE genealogie ADD CONSTRAINT genealogie_pk PRIMARY KEY ( no_enfant );

ALTER TABLE genealogie
  ADD CONSTRAINT genealogie_genealogie_fk FOREIGN KEY ( no_papa )
  REFERENCES genealogie ( no_enfant );
```

Questions :

- 1- Pourquoi est-ce que le no_papa ne doit pas être à NOT NULL ? Donner toutes les explications.
- 2- Peut-on mettre cet attribut à NOT NULL ?
- 3- Insérer des enregistrements (en commençant par vous, vos frères, puis votre père et ses frères, puis le père de votre grand-père Jusqu'à votre 5^{ème} arrière-grand-père
- 4- Écrire les requêtes qui affichent de manière hiérarchique :
 - a. La liste des enfants de votre arrière-grand-père.
 - b. Les parents de votre arrière-grand-père.

Exercice 5, (les fonctions SQL)

Utilisez le document SQL sur le site (**Chapitre 12**)

<http://salihayacoub.com/420Ked/Semaine%202/OracleSQL.pdf>

ou

Utilisez le site d'Oracle à

<https://docs.oracle.com/database/121/SQLRF/functions.htm#SQLRF006>

Choisir 5 fonctions parmi les suivantes, puis répondre aux questions :

1. Donner le rôle des fonctions que vous avez choisi.
2. Donner des exemples d'utilisation.
3. Donner le code SQL qui permet de voir que vous avez compris.

Voici la liste des fonctions :

- DECODE
- TO_DATE
- TO_CHAR
- ADD_MONTHS
- CEIL
- FLOOR
- UPER
- ROUND
- ||
- LPAD

Exercice 6 : (Les séquences)

1. Créer la table EmployesBidon qui va avoir les colonnes suivantes :

Colonnes	Types et contraintes
Emp_no	Number(4,0) primary key
Nom_emp	Varchar2(30) not null
Prenom_emp	Varchar2(30) not null
Salaire_emp	Number(8,2)

2. Créer la séquence SEQEMP.
 - a. La séquence commence à 1
 - b. La séquence s'incrémente de 1
3. Insérer 5 enregistrements de votre choix. Utiliser la séquence pour que le numéro d'employé s'incrémente automatiquement.
4. Pourquoi est-ce que la séquence SEQEMP ne doit pas avoir la propriété CYCLE ?